

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky

Mobilní aplikace pro získávání razítek k věrnostním kartičkám

Petr Čermák

Bakalářská práce  
2024

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2023/2024

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	<b>Petr Čermák</b>
Osobní číslo:	<b>I21132</b>
Studijní program:	<b>B0688A140009 Informační technologie</b>
Téma práce:	<b>Mobilní aplikace pro získávání razítek k věrnostním kartičkám</b>
Zadávající katedra:	<b>Katedra informačních technologií</b>

## Zásady pro vypracování

Cílem práce je vytvoření mobilní aplikace, která běžným uživatelům umožní získávat razítka k věrnostním kartičkám od gastronomických podniků. Podniky budou moci přidat a upravit svůj podnik a také přidávat nové věrnostní kartičky a uživatele do týmu podniku. Pro přidávání razítek bude sloužit mechanismus založený na QR kódech. V teoretické části student popíše problematiku věrnostních kartiček, existující alternativy a technologii QR kódů. V praktické části student navrhne a implementuje mobilní aplikaci s částí pro gastronomické podniky a s částí pro zákazníky. Student bude muset navrhnout aplikaci tak, aby šla jednoduše používat a také tak, aby nešlo podvádět nebo obchodovat s věrnostními razítky mezi uživateli apod. Mobilní aplikace bude vytvořena pomocí jazyku C# a frameworku .NET MAUI.

Rozsah pracovní zprávy: **min. 30 stran**  
Rozsah grafických prací:  
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

BORY, Pavel. C# bez předchozích znalostí. 2. vydání. V Brně: Computer Press, 2022. ISBN 978-80-251-5061-0.  
GOLDMAN, Matt. .NET MAUI in Action. Manning, 2023. ISBN 1633439402.

Vedoucí bakalářské práce: **Ing. Jan Merta, Ph.D.**  
Katedra softwarových technologií

Datum zadání bakalářské práce: **15. prosince 2023**  
Termín odevzdání bakalářské práce: **10. května 2024**

**Ing. Zdeněk Němec, Ph.D.** v.r.  
děkan

L.S.

**Ing. Jan Panuš, Ph.D.** v.r.  
vedoucí katedry

V Pardubicích dne 28. února 2024

## **Prohlášení autora**

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 1. 5. 2024

Petr Čermák

## **Poděkování**

Tímto bych chtěl poděkovat Ing. Janu Mertovi, Ph.D. za jeho odborné vedení a cenné rady. Dále také děkuji svým rodičům, kteří mě podporovali během mého studia.

## **Anotace**

Cílem práce je vytvoření mobilní aplikace, která běžným uživatelům umožní získávat razítka k věrnostním kartičkám od gastronomických podniků. Podniky budou moci přidat a upravit svůj podnik, vytvářet nové věrnostní kartičky a přidávat uživatele do týmu podniku. Pro přidávání razítek bude sloužit mechanismus založený na QR kódech. V teoretické části bude popsána problematika věrnostních kartiček, existující alternativy a technologie QR kódů. V praktické části bude navrhována a implementována mobilní aplikace s částí pro gastronomické podniky a s částí pro zákazníky. Mobilní aplikace bude vytvořena pomocí jazyku C# a frameworku .NET MAUI.

## **Klíčová slova**

Mobilní aplikace, věrnostní programy, QR kódy, gastronomické podniky, .NET MAUI, C#, Android, Firebase

## **Title**

Mobile app for getting stamps for loyalty cards

## **Annotation**

The purpose of this work is to create a mobile application that allows ordinary users to get stamps for loyalty cards from catering businesses. Businesses will be able to add and edit their business, create new loyalty cards, and add users to the business team. A QR-code-based mechanism will be used to add stamps. The theoretical part will describe loyalty cards, existing alternatives, and QR code technology. In the practical part, a mobile application will be designed and implemented, with a part for catering businesses and a part for customers. The mobile application will be created using the C# language and the .NET MAUI framework.

## **Keywords**

Mobile applications, loyalty programs, QR codes, catering businesses, .NET MAUI, C#, Android, Firebase

## Obsah

<b>Seznam zkratk.....</b>	<b>8</b>
<b>Seznam obrázků.....</b>	<b>9</b>
<b>Úvod .....</b>	<b>10</b>
<b>1 Věrnostní karty a existující alternativy .....</b>	<b>11</b>
1.1 Fyzické věrnostní karty .....	11
1.2 Digitální věrnostní karty .....	11
1.3 O věrnostních programech.....	12
1.4 Současné formy věrnostních programů .....	12
1.4.1 Bodové programy .....	12
1.4.2 Členské programy.....	12
1.5 Frekvenční programy.....	12
1.6 Výzvy a omezení současných systémů věrnostních programů .....	13
1.6.1 Nákladnost programů .....	13
1.6.2 Složitost .....	13
1.6.3 Saturace trhu .....	13
1.7 Alternativy .....	13
1.7.1 Stammgast .....	13
1.7.2 Con Gusto .....	14
1.7.3 Qerko .....	15
<b>2 Technologie QR kódů.....</b>	<b>17</b>
2.1 Funkcionalita QR kódů.....	17
2.2 Využití QR kódů.....	17
2.3 Výhody a výzvy QR kódů .....	17
<b>3 Technologie použité ve vývoji.....</b>	<b>18</b>
3.1 Jazyk C# .....	18
3.1.1 Architektura .NET .....	18
3.1.2 Kompilace a spouštění.....	18
3.2 Jazyk XAML .....	18
3.3 Model MVVM.....	19
3.4 .NET MAUI.....	20
3.4.1 Platformová podpora a API .....	20

3.4.2	Klíčové funkce a výhody .....	20
3.5	Visual Studio .....	20
3.5.1	Funkce a možnosti sady Visual Studio.....	20
3.5.2	Instalace a konfigurace .....	21
3.6	Firebase.....	21
3.6.1	Firebase Authentication .....	21
3.6.2	Firebase Cloud Storage.....	21
3.6.3	Cloud Firestore .....	22
<b>4</b>	<b>Návrh mobilní aplikace.....</b>	<b>23</b>
4.1	Funkční požadavky.....	23
4.2	Nefunkční požadavky .....	25
<b>5</b>	<b>Vývoj mobilní aplikace.....</b>	<b>26</b>
5.1	Vzhled aplikace .....	26
5.2	Registrace a přihlášení.....	26
5.2.1	Registrace .....	26
5.2.2	Přihlášení .....	28
5.3	Správa dat a session management .....	29
5.3.1	Třída SessionManager .....	29
5.3.2	Třída DataManager.....	29
5.4	Hlavní obrazovka.....	29
5.5	Uživatelský profil .....	30
5.5.1	Věrnostní karty, které má uživatel orazítkované .....	31
5.5.2	Podniky přihlášeného uživatele .....	31
5.5.3	Změna údajů .....	32
5.5.4	Nastavení .....	32
5.6	Obrazovka podniku .....	33
5.6.1	Informace o podniku.....	33
5.6.2	Věrnostní karty .....	33
5.6.3	Zaměstnanci.....	34
5.6.4	Přidávání a editace podniku.....	34
5.7	Využití služby Firebase v mobilní aplikaci .....	34
5.7.1	Firebase Authentication v aplikaci .....	34
5.7.2	Firebase Cloud Storage v aplikaci .....	34



5.7.3 Cloud Firestore v aplikaci .....	35
5.8 Generování a skenování QR kódů .....	35
<b>Závěr .....</b>	<b>40</b>
<b>Literatura .....</b>	<b>41</b>

## Seznam zkratek

AES	Advanced Encryption Standard
API	Application Programming Interface
CLI	Common Language Infrastructure
CLR	Common Language Runtime
CTS	Common Type Specification
IDE	Integrated Development Environment
IL	Intermediate Language
JIT	Just-In-Time
MAUI	Multi-platform App UI
MVVM	Model-View-ViewModel
NFC	Near Field Communication
NoSQL	non-SQL
QR	Quick Response
SDK	Software Development Kit
SMS	Short Message Service
VIP	Very Important Person
XAML	Extensible Application Markup Language
ZXing	Zebra Crossing

## Seznam obrázků

Obrázek 1 – Stammgast věrnostní karta v peněženke (zdroj: [8]).....	14
Obrázek 2 – Fyzická věrnostní karta Con Gusto (zdroj: [11]) .....	15
Obrázek 3 – Qerko platba (zdroj: [12]) .....	15
Obrázek 4 – QR s odkazem na webovou stránku Univerzity Pardubice (zdroj: vlastní) ....	17
Obrázek 5 – Návrhový vzor MVVM (zdroj: [19]) .....	19
Obrázek 6 – Diagram případu užití pro mobilní aplikaci (zdroj: vlastní) .....	24
Obrázek 7 – Obrazovka pro registraci (zdroj: vlastní) .....	27
Obrázek 8 – Příklad potvrzovací e-mailové zprávy (zdroj: vlastní).....	28
Obrázek 9 – Obrazovka pro přihlášení (zdroj: vlastní) .....	28
Obrázek 10 – Příklad vzhledu karty pro jednotlivý podnik (zdroj: vlastní).....	30
Obrázek 11 – Ukázka části profilu s horizontálním posuvníkem (zdroj: vlastní).....	30
Obrázek 12 – Věrnostní karta obsahující razítko (zdroj: vlastní).....	31
Obrázek 13 – Zobrazení profilového obrázku s možností nahrát nový (zdroj: vlastní).....	32
Obrázek 14 – Možnost skenování a úprav pro oprávněné uživatele (zdroj: vlastní) .....	33
Obrázek 15 – Firebase Cloud Storage s ukázkou unikátních názvů (zdroj: vlastní).....	35
Obrázek 16 – Metoda využívaná pro zašifrování dat pomocí šifry AES (zdroj: vlastní) ...	36
Obrázek 17 – Diagram aktivit ukazující průběh generování QR kódů (zdroj: vlastní).....	36
Obrázek 18 – Zvětšený, vygenerovaný QR kód (zdroj: vlastní) .....	37
Obrázek 19 – Diagram aktivit ukazující průběh skenování QR kódů (zdroj: vlastní) .....	38
Obrázek 20 – Chybová hláška pro zneplatněný QR kód (zdroj: vlastní) .....	39

## Úvod

V éře rychlého technologického rozvoje se chytré telefony staly neodmyslitelnou součástí každodenního života. Díky své schopnosti přístupu a uchování informací na jednom kompaktním zařízení, přinášení chytré telefony uživatelům řadu výhod. Jednou z oblastí, kde moderní technologie mohou nacházet uplatnění, jsou věrnostní programy nabízené gastronomickými podniky. Tyto programy tradičně využívají fyzické karty pro sběr razítek, což může být pro zákazníky i podniky značně nepraktické. Fyzické karty navíc přinášejí zákazníkům řadu nevýhod, jako je riziko ztráty nebo zapomenutí. Ztráta nebo zapomenutí karet může vést k nespokojenosti zákazníků, zatímco pro podniky znamená ztrátu příležitosti pro budování vztahu s klienty a zvýšení jejich loajality.

Tato bakalářská práce se zaměřuje na integraci věrnostního systému do mobilní aplikace s cílem zjednodušit a zefektivnit používání věrnostních karet. Aplikace je navržena tak, aby vyhovovala potřebám jak majitelů gastronomických podniků, tak běžných uživatelů, kteří chtějí mít všechny své věrnostní karty přehledně uspořádané ve svém chytrém telefonu. Kromě toho aplikace umožňuje manažerům podniků přidávat a spravovat informace o svých podnicích a nabízet uživatelům věrnostní karty, které jim umožňují získávat odměny za jejich plnění.

Aplikace je navržena speciálně pro uživatele chytrých telefonů s operačním systémem Android, který si v roce 2024 udržuje dominantní postavení na globálním trhu s mobilními zařízeními. S celosvětovým tržním podílem 69,88 %, Android převyšuje všechny ostatní operační systémy ve své kategorii. Na českém trhu je jeho postavení ještě silnější s podílem 73,42 %. Pro srovnání, operační systém iOS od společnosti Apple drží na světovém trhu podíl 29,39 %, zatímco na českém trhu dosahuje 26,05. [1], [2]

Ačkoliv framework .NET MAUI, který je využíván pro vývoj této aplikace, podporuje i operační systém iOS, rozhodl jsem se zaměřit na platformu Android. Tento výběr byl částečně motivován mou osobní zkušeností a vlastnictvím zařízení s tímto systémem, což mi umožňuje lépe testovat a optimalizovat aplikaci. Dalším důvodem pro zaměření na Android je jeho silné postavení na trhu, což nabízí větší dosah a potenciální uživatelskou základnu pro aplikaci.

Teoretická část se zaměřuje na věrnostní karty, přičemž rozlišuje mezi fyzickými a digitálními kartami. Dále poskytuje podrobnou analýzu současných věrnostních systémů, nabízí přehled stávajících alternativ a zkoumá technologie QR kódů, které jsou klíčové pro efektivní správu a bezpečné uchování dat v digitálních věrnostních kartách. Výběr tohoto tématu byl motivován také relativním nedostatkem aplikací na českém trhu, které by integrovaly věrnostní systémy založené na razítkách do digitální formy. Cílem této práce není pouze navrhnout funkční a uživatelsky přívětivé řešení, ale také analyzovat potenciální přínosy a výzvy spojené s digitalizací věrnostních systémů.

## **1 Věrnostní karty a existující alternativy**

V posledních letech došlo k výrazným změnám v tom, jak fungují věrnostní programy. V minulosti byly věrnostní programy často realizovány pomocí fyzických karet, jako jsou razítkové karty nebo členské karty, zatímco dnes dominují digitální věrnostní karty. Digitální karty, integrované do mobilních aplikací, nabízejí uživatelům větší pohodlí a flexibilitu. Umožňují snadné spravování a využívání věrnostních benefitů přímo z mobilního telefonu, což se stává stále důležitějším v éře, kdy lidé stále více používají své smartphony pro každodenní činnosti. [3]

### **1.1 Fyzické věrnostní karty**

Fyzické věrnostní karty jsou mezi uživateli známé, snadno se zavádějí a jednoduše se používají. Přestože zřízení tradičního věrnostního programu s fyzickými kartami může na první pohled působit jako cenově dostupná možnost, skrývá několik významných nevýhod. Jednou z hlavních komplikací je, že tyto karty mohou být snadno zapomenuty nebo ztraceny. [4]

Fyzické věrnostní karty představují významné náklady spojené s tiskem. Je nutné mít vždy dostatečný počet karet pro všechny zákazníky, kteří tyto karty chtějí využít. Může jít nejen o nové zákazníky, ale také o ty, kteří svou kartu ztratili nebo zapomněli, a o zákazníky, jejichž karta je již plně vyplněná a nelze ji znovu použít. Podniky se také musí vyrovnat se stížnostmi zákazníků, kteří si stěžují, že kartu zapomněli nebo ztratili. [4]

Dalším problémem je nemožnost efektivního sledování návratnosti investic věrnostního programu. Tradiční metody, jako jsou fyzické karty pro získání razítek, se obtížně sledují, a nelze s jistotou říct, jak účinný je věrnostní program. [4]

### **1.2 Digitální věrnostní karty**

Digitální věrnostní karty nabízí oproti fyzickým kartám řadu výhod. Jsou ekologičtější, snadno spravovatelné a zákazníci je nemohou ztratit, protože jsou uloženy přímo v jejich mobilních zařízeních. Díky propojení s online účty zákazníků mohou podniky získat cenné informace o nákupních zvycích zákazníků a na základě toho přizpůsobovat nabídky. Digitalizace rovněž umožňuje okamžité přidělování bodů do věrnostního programu. [4]

Digitální karty a programy jsou budoucností věrnostních programů, které se rychle přizpůsobují technologickému pokroku a umožňují podnikům zůstat v kroku s novými trendy. [4]

Z uvedených informací je zřejmé, že digitální věrnostní karty nabízejí více výhod než karty fyzické. Je zde, ale pořád spousta zákazníků, kteří preferují standardní karty od fyzických, a proto je potřeba zvážit jaké karty jsou pro daný podnik výhodnější.

### **1.3 O věrnostních programech**

Věrnostní programy jsou strategie, které používají podniky k odměňování zákazníků za jejich pravidelné nákupy. Tato praxe není novinkou; sahá až do doby velké hospodářské krize, kdy společnost Betty Crocker nabídla zákazníkům možnost zasílat vršky krabiček od jejich produktů výměnou za kupóny na další nákupy. Od té doby věrnostní programy prošly dlouhou cestou a dnes jsou významně podpořeny moderními technologiemi. [5]

Věrnostní programy jsou obzvláště cenné pro restaurace a jiné služby zaměřené na spotřebitele, protože úspěšné programy mohou motivovat zákazníky k častějším návštěvám a vyšším výdajům, než by jinak uskutečnili. Věrnostní programy mohou přinášet mnoho výhod, od přilákání nových zákazníků, přes proměnu příležitostných návštěvníků na pravidelné hosty, až po zvýšení celkových tržeb. [5]

Existuje několik způsobů, jak mohou být věrnostní programy strukturovány, od jednoduchých karet na razítka až po složitější systémy založené na bodech nebo placené členství, které nabízí VIP výhody. Bez ohledu na formu mají dobře spravované věrnostní programy stejný cíl, kterým je zvyšovat spokojenost zákazníků, prohlubovat jejich vztah s obchodem a posilovat jejich loajalitu. [5]

### **1.4 Současné formy věrnostních programů**

Věrnostní programy se vyskytují v mnoha variantách, přičemž každá nabízí specifické vlastnosti a přínosy jak pro zákazníky, tak pro podniky. Mezi příklady patří bodové, členské a frekvenční programy.

#### **1.4.1 Bodové programy**

Bodové programy jsou jedním z nejpobulárnějších typů věrnostních programů. Zákazníci získávají body za každý nákup, které mohou později vyměnit za slevy, zboží nebo jiné výhody. Tento typ programu motivuje zákazníky k opakovaným nákupům, protože chtějí získat dostatek bodů pro určité odměny. [5], [6]

#### **1.4.2 Členské programy**

Některé podniky nabízejí věrnostní programy, které vyžadují placené členství výměnou za pravidelné výhody, jako jsou exkluzivní slevy nebo přístup k speciálním akcím. [5]

Tyto programy často přinášejí vyšší vnímanou hodnotu a mohou efektivně vytvářet pocit exkluzivity a příslušnosti.

### **1.5 Frekvenční programy**

Ve frekvenčních programech, je jedno kolik peněz zákazník utratí nebo kolik získal bodů. Zákazníci jsou odměňováni za četnost svých návštěv. [5]

## **1.6 Výzvy a omezení současných systémů věrnostních programů**

Věrnostní programy přinášejí řadu výzev a omezení, která mohou komplikovat jejich efektivní implementaci a správu.

### **1.6.1 Nákladnost programů**

Implementace efektivního věrnostního programu může vyžadovat významné finanční investice. Vysoké slevy nebo bohaté odměny mohou zatížit rozpočet firmy, a i když je zpětná vazba dobrá, neexistuje záruka zisku. Je proto klíčové pečlivě navrhnout program tak, aby podporoval opakované nákupy a současně kompenzoval veškeré náklady. [7]

### **1.6.2 Složitost**

Správné nastavení věrnostního programu může být časově náročné a jeho komplexnost může potenciálně odradit zákazníky. Je důležité, aby byl program snadno pochopitelný; studie ukázaly, že složitý registrační proces může způsobit, že zákazníci od programu odstoupí. Celý systém by měl být intuitivní, transparentní a skutečně atraktivní pro účastníky. [7]

### **1.6.3 Saturace trhu**

Trh je zaplaven množstvím věrnostních programů nabízených různými podniky, které často nabízejí podobné benefity a strukturu. Klíčové je vytvořit věrnostní program, který je jedinečný a odráží charakter značky. [7]

Správné pochopení předností a nedostatků věrnostních programů je nezbytné pro vývoj efektivního systému, který přinese hodnotu jak zákazníkům, tak firmě. V kontextu rostoucí konkurence je nezbytné, aby byl program vnímán jako unikátní. To vyžaduje důkladné plánování a pečlivě promyšlenou strategii, která respektuje hodnoty značky a zároveň ji odlišuje od konkurence. [7]

## **1.7 Alternativy**

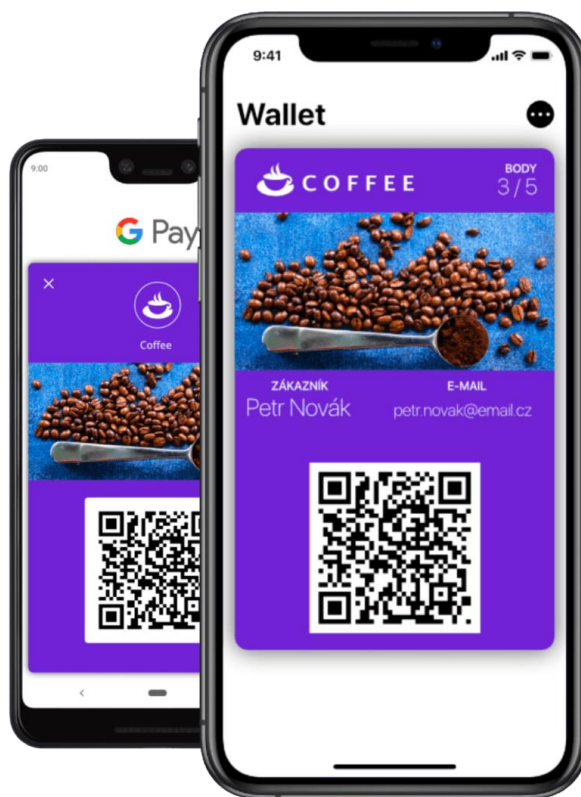
Ve světě existuje spousta aplikací určených pro správu věrnostních programů, ale na českém trhu jich není dostupných mnoho. V této kapitole jsou popsány tři již existující aplikace, které podporují nějakým způsobem věrnostní programy a jsou dostupné na českém trhu.

### **1.7.1 Stammgast**

Stammgast je věrnostní program speciálně určený pro malé podniky. Tento systém umožňuje podnikům vytvořit digitální kartu, která působí jako malá webová stránka firmy. Na kartě mohou být zobrazeny informace o nabídce, kontaktech nebo otevíracích hodinách, a také aktuální stav zákaznického účtu a identifikační QR kód. [8]

Pro využití digitální karty vytvořené skrze Stammgast nemusí zákazník stahovat žádnou aplikaci. Zákazníci si mohou kartu stáhnout a uložit do své mobilní peněženky, jako je Apple Wallet nebo Google Pay. Stačí naskenovat QR kód, který jim podnik nabídne, a karta je aktivní. Zákazník může ihned začít sbírat body za věrnost. Při placení zákazník ukáže kartu v telefonu, zaměstnanec podniku naskenuje QR kód a v aplikaci se zobrazí

zákaznické konto, kde lze přidat nové body nebo zaznamenat využití již získaných odměn. [8], [9]



Obrázek 1 – Stammgast věrnostní karta v peněženke (zdroj: [8])

Stammgast vyžaduje formu předplatného. Tento model předplatného zahrnuje pravidelné měsíční nebo roční platby, které zajišťují provoz a podporu digitálních věrnostních karet a souvisejících nástrojů. [8]

Předplatné je nevýhoda této služby, může být finančně náročné pro některé malé podniky, kteří chtějí zavést nebo udržovat věrnostní program bez značných počátečních investic.

### 1.7.2 Con Gusto

Con Gusto je síť brněnských gastronomických podniků, která nabízí věrnostní program integrující pokladní systémy restaurací s aplikací na rozvoz jídla. Po registraci na webové stránce Con Gusto obdrží zákazník digitální věrnostní kartu, kterou si může uložit do svého mobilního zařízení nebo si nechat zaslat ve fyzické podobě. Věrnostní systém automaticky spáruje platby se zákaznickým účtem a připíše na něj bonusy ve formě vrácení peněz. Kromě toho, pokud má zákazník dostatečný kredit, může ho využít k částečné nebo úplné úhradě svého účtu. [10], [11]



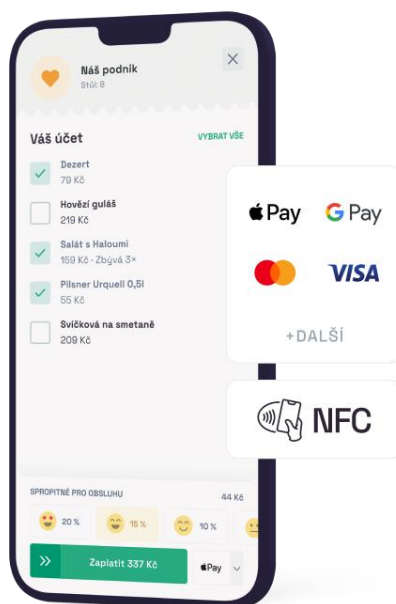


Obrázek 2 – Fyzická věrnostní karta Con Gusto (zdroj: [11])

Věrnostní program Con Gusto lze využít pouze v podnicích, které jsou součástí této sítě. Pro integraci tohoto systému je tedy nutné, aby podnik byl členem sítě Con Gusto. [11]

### 1.7.3 Qerko

Qerko je aplikace, která je určena pro gastronomické podniky, které chtějí svým zákazníkům nabídnout pohodlné a rychlé řešení pro platbu a objednávání přímo z jejich chytrých telefonů. Aplikace využívá QR kódy, umožňující zákazníkům prohlížet menu, objednávat jídlo a nápoje, a dokonce i platit za své objednávky bez potřeby čekání na obsluhu. Hosté nemusí stahovat aplikaci Qerko pro platby, po naskenování QR kódu nebo načtení přes NFC čip se jim stáhne tzv. „dočasná aplikace“, která se po ukončení smaže. [12]



Obrázek 3 – Qerko platba (zdroj: [12])

Kromě zjednodušení objednávek a plateb, Qerko nabízí i integrovaný věrnostní program. Tento program umožňuje zákazníkům sbírat body za každou transakci provedenou prostřednictvím aplikace, které pak mohou vyměnit za slevy, zdarma produkty, nebo jiné výhody. [13]

Nicméně, pro získání odměny za věrnost je potřeba platit Qerkem [14]. Nutnost platit Qerkem pro získání odměn za věrnost, může omezovat zákazníky preferující jiné formy platby. V takovém případě by se zákazník musel obrátit na obsluhu a informovat se o alternativních možnostech získání věrnostních bodů. Toto omezení může být pro některé zákazníky nevýhodou, jelikož komplikuje proces sběru bodů, pokud nechtějí nebo nemohou využít platbu přes Qerko.

Další potenciální nevýhodou je, že Qerko si účtuje poplatky za každou platbu provedenou skrze aplikaci [12]. Qerko může podnikům vydělat více peněz, protože se zrychlí celý proces platby a podniky jich dokážou obsloužit více. Přesto mohou transakční poplatky ovlivnit podniky při jejich rozhodování o integraci této technologie.

## 2 Technologie QR kódů

QR kód neboli Quick Response kód, v češtině můžeme říct „rychlá reakce“, je typ dvoudimenzionálního čárového kódu, který byl vyvinut v 90. letech firmou Denso Wave, dceřinou společností Toyota, pro sledování vozidel během výroby. QR kódy jsou schopny uchovávat větší množství dat než tradiční čárové kódy a lze je snadno skenovat digitálními zařízeními, jako jsou mobilní telefony. [15]



Obrázek 4 – QR s odkazem na webovou stránku Univerzity Pardubice (zdroj: vlastní)

QR kódy byly v této aplikaci zvoleny pro jejich snadnou implementaci a spolehlivost.

### 2.1 Funkcionalita QR kódů

QR kódy obvykle obsahují vzory uspořádané v mřížce na kontrastním pozadí, nejčastěji černé na bílém, ale mohou být také vytvořeny v jiných barvách. Data jsou zakódována v binárním formátu a mohou být reprezentována v několika režimech, kterými jsou alfanumerický, číselný, binární a Kanji. Při skenování QR kódu nezáleží na úhlu kamery, mělo by se podařit získat informace z jakéhokoliv úhlu. [15], [16]

### 2.2 Využití QR kódů

QR kódy nacházejí uplatnění v mnoha oblastech od marketingu, přes správu zásob, až po digitální platby a kryptoměny. Jsou často využívány pro rychlé sdílení informací, jako jsou webové adresy, informace o produktech, digitální menu v restauracích, nebo dokonce kryptoměnové adresy. Během pandemie COVID-19 došlo k nárůstu využití QR kódů, zejména v restauracích a hotelích, pro bezdotykovou komunikaci. [15]

### 2.3 Výhody a výzvy QR kódů

QR kódy umožňují snadné a rychlé získávání informací, což vede k lepší angažovanosti zákazníků a zefektivnění operací. Mezi potenciální nevýhody však patří obavy o bezpečnost, protože mohou odkazovat na škodlivé webové stránky. Uživatelé by měli skenovat pouze QR kódy z důvěryhodných zdrojů. [15]

## 3 Technologie použité ve vývoji

### 3.1 Jazyk C#

C# je moderní, objektově orientovaný a typově bezpečný programovací jazyk, který je součástí platformy .NET. Tento jazyk, známý pro svou schopnost vývoje zabezpečených a robustních aplikací, má své kořeny v rodině jazyků C. Pro programátory s předchozími zkušenostmi v C, C++, Java a JavaScript je C# hned povědomý, což usnadňuje přechod z těchto jazyků na jazyk C#. [17]

Rozhodnutí pro využití jazyka C# pro tento projekt bylo motivováno zkušenostmi s tímto jazykem, což umožnilo efektivní využití jeho pokročilých funkcí pro vývoj této aplikace.

#### 3.1.1 Architektura .NET

Programy v C# fungují na platformě .NET, která využívá virtuální spouštěcí systém CLR a nabízí bohatou sadu třídních knihoven. CLR je implementací CLI a podporuje interoperabilitu mezi různými programovacími jazyky. [17]

Jednou z hlavních vlastností .NET je jeho schopnost integrace kódů napsaných v různých jazycích, jako jsou F#, Visual Basic nebo C++. Tyto jazyky generují kód IL, který odpovídá specifikaci CTS. Díky tomu mohou různé moduly spolupracovat, jako kdyby byly napsány ve stejném programovacím jazyce. Knihovny, které .NET zahrnuje, poskytují podporu pro široké spektrum programátorských úloh, od souborového vstupu a výstupu po analýzu XML, vývoj webových aplikací a Windows Forms. [17]

#### 3.1.2 Kompilace a spouštění

Zdrojový kód v C# je zkompileován do kódu IL. Tento IL kód je uložen v sestavení s příponou .dll, společně s dalšími zdroji, jako jsou rastrové obrázky a řetězce. Sestavení také obsahuje manifest, který poskytuje informace o typech, verzi a jazykové verzi sestavení. Když je program spuštěn, CLR načte toto sestavení a provede JIT kompilaci, při které je IL kód převeden na nativní instrukce počítače. [17]

### 3.2 Jazyk XAML

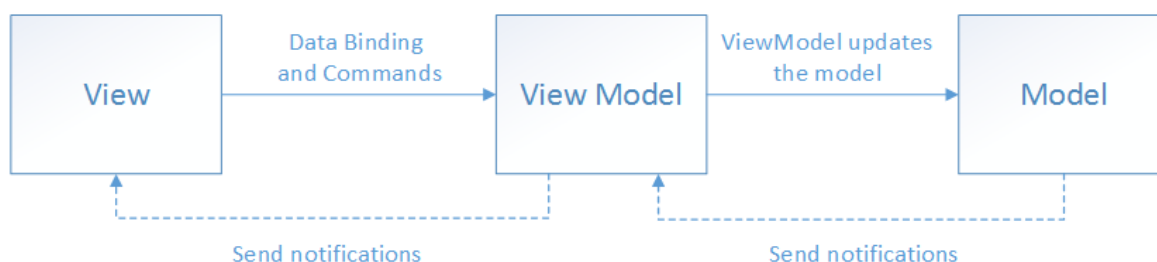
XAML je jazyk založený na XML, který se využívá pro definování uživatelských rozhraní ve frameworku .NET MAUI. XAML je v podstatě XML, ale má několik jedinečných funkcí. Tento jazyk umožňuje vývojářům navrhovat rozhraní pomocí deklarativních značek namísto procedurálního kódu. Není povinné využívat XAML v aplikaci založené na .NET MAUI, ale doporučuje se ho využívat při vývoji uživatelského rozhraní, protože je často vizuálně přehlednější a stručnější. Mezi unikátní funkce patří elementy vlastností, připojené vlastnosti a rozšíření značek. [18]

XAML je ideální pro implementaci designového vzoru MVVM, kde se zobrazení a logika aplikace oddělují. V tomto případě XAML tvoří zobrazení, které je spojené s logikou modelu prostřednictvím datových vazeb založených na XAML. [18]

Nicméně XAML má i svá omezení. Neumožňuje obsahovat programový kód, což znamená, že veškeré obslužné rutiny událostí musí být implementovány v doprovodných souborech kódu. [18]

### 3.3 Model MVVM

Model MVVM je návrhový vzor, který rozděluje aplikaci na tři hlavní komponenty. Tyto komponenty zahrnují Model, View a ViewModel. Tento vzor umožňuje efektivní oddělení prezentace informací od logiky aplikace. [19]



**Obrázek 5 – Návrhový vzor MVVM (zdroj: [19])**

View je zodpovědné za definici struktury a vzhledu uživatelského rozhraní, které vidí uživatel. V ideálním případě by mělo být View oddělené od ostatních vrstev pomocí XAML. [19]

ViewModel funguje jako most mezi View a Modelem. Poskytuje data a příkazy, ke kterým může být View vázáno, a reaguje na akce uživatele zobrazením změn. ViewModel také zprostředkovává data z Modelu a mění je na formáty, které jsou vhodnější pro View, což usnadňuje data binding. [19]

Model obsahuje nebo zpracovává data aplikace a obchodní logiku. Je to část aplikace, kde jsou definovány všechny datové operace, pravidla, obchodní logika a další funkcionality, která je nezávislá na uživatelském rozhraní. [19]

Výhody použití MVVM zahrnují lepší oddělení a nezávislost komponent, což usnadňuje údržbu a testování aplikace. ViewModel podporuje jednoduché jednotkové testování, protože nezávisí na uživatelském rozhraní a lze ho testovat bez View. Navíc umožňuje designérům a vývojářům pracovat paralelně, přičemž každý se může soustředit na své specifické komponenty. [19]

### 3.4 .NET MAUI

.NET MAUI je moderní framework umožňující vývoj multiplatformních aplikací s jednou kódovou základnou pro Windows, Android, macOS a iOS. Je to evoluce Xamarin.Forms, navržená pro poskytování bohatšího uživatelského rozhraní a lepšího výkonu na mobilních i desktopových platformách. [20]

.NET MAUI byl pro tento projekt zvolen z důvodu využití jazyka C# a mého zájmu o využití nejmodernějšího dostupného frameworku.

#### 3.4.1 Platformová podpora a API

.NET MAUI integruje rozhraní API pro Android, iOS, macOS a Windows do jednotného rozhraní API. Toto sjednocené API umožňuje vývojářům pracovat s jednotným kódem pro všechny platformy s dodatečnou podporou všech aspektů nativních platforem. Aplikace lze kompilovat do nativních balíčků pro jednotlivé platformy, což zahrnuje speciální případy jako Mac Catalyst pro macOS nebo WinUI 3 pro Windows. [20]

Framework rovněž poskytuje rozsáhlé rozhraní API pro nativní funkce zařízení, což umožňuje aplikacím využívat senzory jako je akcelerometr, kompas a gyroskop, monitorovat a reagovat na změny síťového připojení, získávat podrobné informace o zařízení, a umožňuje interakci s textem a soubory mezi aplikacemi. Kromě toho API podporuje bezpečné ukládání dat, převod textu na řeč a zahajování autentizačních toků. [20]

#### 3.4.2 Klíčové funkce a výhody

.NET MAUI nabízí kolekci ovládacích prvků, podporu datových vazeb, rozhraní API pro přístup k nativním funkcím zařízení, grafické funkce a projektový systém, který umožňuje cílení na více platforem. Zahrnuje také funkce pro opětovné načítání .NET za provozu, což umožňuje upravovat zdrojový kód během běhu aplikace. Tyto změny jsou poté zobrazeny ve spuštěné aplikaci bez nutnosti rekompilace. .NET MAUI také podporuje opětovné načítání XAML za provozu, změny provedené zde se rovněž projeví ve spuštěné aplikaci bez nutnosti opětovné kompilace. [20]

### 3.5 Visual Studio

Visual Studio je komplexní integrované vývojové prostředí (IDE), které poskytuje vývojářům nástroje a funkce pro kompletní vývojový cyklus aplikací, od psaní kódu přes jeho ladění až po nasazení. Visual Studio podporuje množství programovacích jazyků a platforem, umožňuje efektivní spolupráci na projektech a integraci správy verzí. Díky svým rozsáhlým funkcím je ideální pro tvorbu, testování a úpravu aplikací pro různé platformy, včetně mobilních a desktopových aplikací v .NET, ASP.NET a C++. [21]

#### 3.5.1 Funkce a možnosti sady Visual Studio

Sada Visual Studio je vybavena nástroji, které zjednodušují vývojářské procesy. Editor kódu a jeho přizpůsobení, uspořádání kódu a návrhové nástroje usnadňují psaní čistého a efektivního kódu. Dále IDE nabízí integrované ladění, které pomáhá diagnostikovat a řešit problémy v kódu, podporuje testování jednotek pro zajištění kvality kódu, a umožňuje

snadné nasazení aplikací na různé platformy. Podpora vývoje s asistencí umělé inteligence, jako jsou GitHub Copilot a IntelliCode, zvyšuje efektivitu psaní kódu tím, že předpovídá potřeby vývojářů a automaticky doplňuje kód. [21]

### **3.5.2 Instalace a konfigurace**

Pro prvotní začátek využívání Visual Studia musí vývojáři nejprve vybrat a stáhnout požadovanou edici IDE, která nejlépe odpovídá jejich potřebám. Existuje několik edicí od bezplatné komunitní verze pro studenty a individuální vývojáře, přes profesionální pro malé týmy, až po enterprise verzi pro velké organizace. [21]

## **3.6 Firebase**

Google Firebase je sada cloudových vývojových nástrojů, která pomáhá vývojářům mobilních aplikací vytvářet, nasazovat a škálovat jejich aplikace. Mezi klíčové funkce, které Firebase poskytuje, patří autentizace uživatelů, cloudové úložiště a NoSQL databáze. [22]

Služba Firebase je v aplikaci využita pro autentizaci, úložiště a NoSQL databázi.

Tyto služby jsou přístupné přes bezplatný plán, který nevyžaduje zadání platebních údajů a po vyčerpání bezplatných kvót se služby automaticky vypnou, což uživatelům eliminuje riziko neočekávaných plateb. [23]

### **3.6.1 Firebase Authentication**

Firebase Authentication poskytuje snadnou a bezpečnou metodu pro správu uživatelských účtů a autentizaci pro aplikace. Tento modul je navržen tak, aby pomáhal vývojářům implementovat všechny běžné formy autentizace, včetně hesel, tokenů z externích poskytovatelů identity (jako jsou Google, Facebook a Twitter), a dokonce i telefonní autentizace pomocí SMS. Kromě toho Firebase Authentication podporuje vícefaktorovou autentizaci, která přidává další vrstvu zabezpečení tím, že vyžaduje více než jeden důkaz totožnosti od uživatelů před poskytnutím přístupu. [24]

Firebase Authentication rovněž nabízí integrovanou funkcionalitu pro zasílání e-mailových zpráv. Uživatelům lze zasílat e-mailové zprávy obsahující konkrétní informace, jako je resetování zapomenutého hesla nebo verifikace e-mailové adresy. [25]

### **3.6.2 Firebase Cloud Storage**

Firebase Cloud Storage nabízí výkonné, bezpečné a škálovatelné řešení pro ukládání souborů a dat v cloudu, které je navrženo speciálně pro potřeby mobilních a webových aplikací. Tato služba umožňuje vývojářům snadno ukládat obrázky, videa a další typy obsahu bez nutnosti spravovat infrastrukturu pro ukládání dat. Integrovaná s robustními bezpečnostními opatřeními, včetně šifrování dat a podrobných pravidel přístupu, zajišťuje, že data jsou chráněna a dostupná pouze pro oprávněné uživatele. Navíc díky automatické škálovatelnosti služba roste s potřebami aplikace, což vývojářům umožňuje bez obav přizpůsobovat kapacitu úložiště aktuálním požadavkům. Cloud Storage je také úzce integrován s ostatními službami Firebase, což umožňuje snadnou synchronizaci

s autentizačními a databázovými službami a poskytuje vývojářům komplexní řešení pro správu aplikací. [26]

### **3.6.3 Cloud Firestore**

Cloud Firestore je flexibilní a škálovatelná cloudově-hostovaná NoSQL databáze, navržená pro synchronizaci dat v reálném čase napříč klientskými aplikacemi. Umožňuje přístup prostřednictvím nativních SDK pro platformy Apple, Android a web, stejně jako prostřednictvím Node.js, Java, Python, Unity, C++ a Go SDK. [27]

Firestore je organizována do kolekcí a dokumentů. Data jsou ukládána do dokumentů, které obsahují pole mapující na hodnoty, a tyto dokumenty jsou seskupeny do kolekcí. Firestore podporuje různé typy dat, od jednoduchých řetězců a čísel až po složité, vnořené objekty. Umožňuje vytvářet vnořené kolekce uvnitř dokumentů a stavět hierarchické datové struktury, které se mohou rozšiřovat podle potřeb databáze. [27]

Zabezpečení dat v Cloud Firestore je primárně zajištěno pomocí Firebase Authentication. ale dá se také zabezpečit pomocí různých bezpečnostních pravidel. [27]



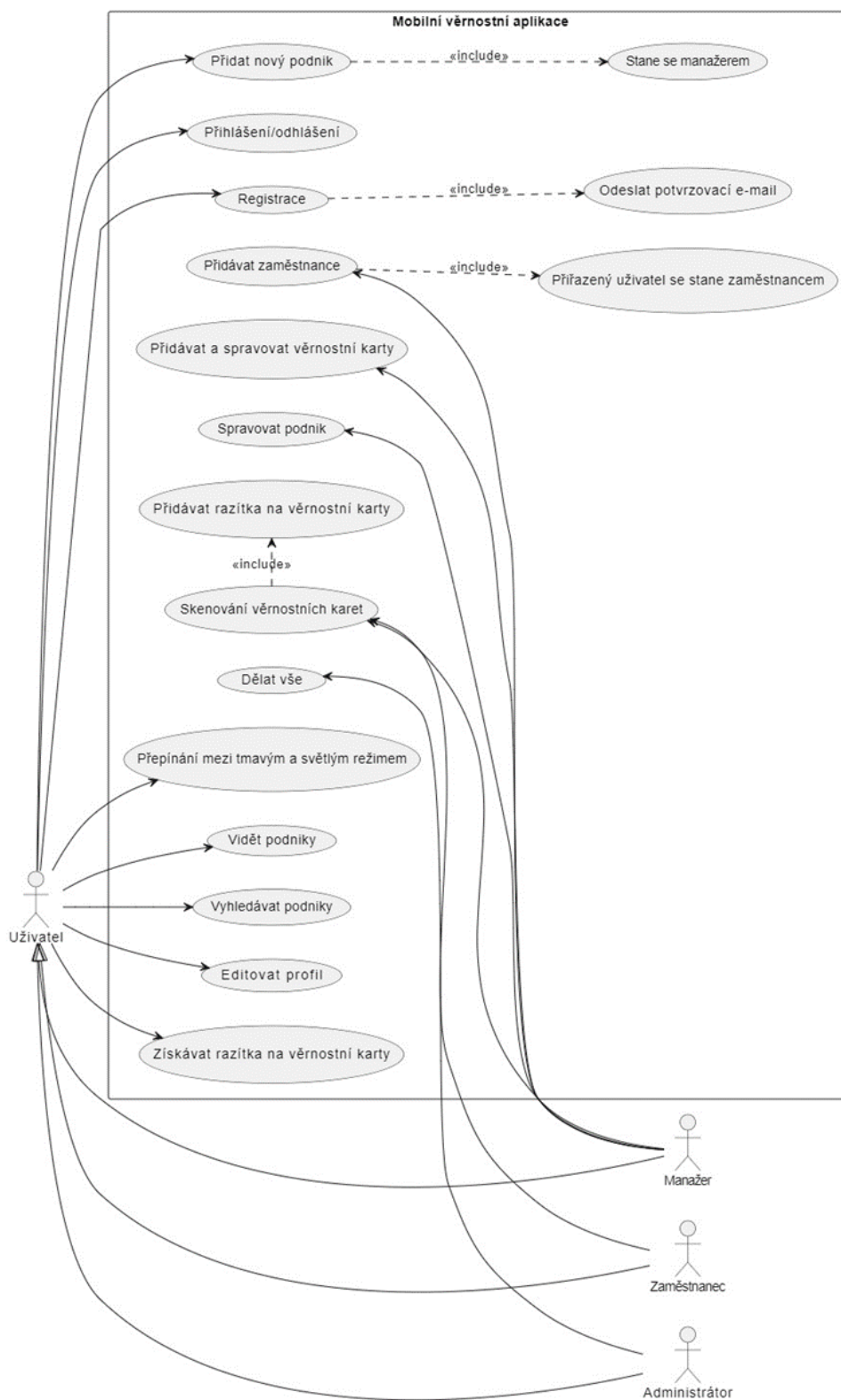
## 4 Návrh mobilní aplikace

Věrnostní mobilní aplikace slouží obyčejným uživatelům ke sbírání věrnostních karet pomocí QR kódů a manažerům podniku k správě podniků a příslušných věrnostních karet. Při návrhu této aplikace bylo nutné analyzovat funkční a nefunkční požadavky, které bude aplikace využívat.

### 4.1 Funkční požadavky

- FP 1: Autentizace uživatelů
  - FP 1.1: Aplikace bude umožňovat registraci a přihlášení uživatelů pomocí e-mailu a hesla.
  - FP 1.2: Aplikace bude automaticky odesílat potvrzovací e-mail po dokončení registrace.
- FP 2: Správa uživatelského profilu
  - FP 2.1: Uživatelé budou moci na svém profilu upravovat své údaje.
    - FP 2.1.1: Uživatelé budou moci upravovat své osobní údaje
    - FP 2.1.2: Uživatelé budou moci měnit svůj avatar.
    - FP 2.1.3: Uživatelé budou moci měnit svůj e-mail a heslo.
  - FP 2.2: Uživatelé se budou moci odhlásit na svém profilu.
- FP 3: Prohlížení a filtrování podniků
  - FP 3.1: Aplikace bude zobrazovat seznam všech podniků.
  - FP 3.2: Uživatelé budou moci filtrovat podniky podle názvu, typu nebo geografické polohy.
- FP 4: Přepínání režimu zobrazení
  - FP 4.1: Uživatelé budou moci přepínat mezi tmavým a světlým režimem zobrazení.
- FP 5: Správa podniků
  - FP 5.1: Uživatelé budou moci přidávat podniky a stanou se jejich manažery.
  - FP 5.2: Manažeři budou moci spravovat podniky a jejich informace.
  - FP 5.3: Manažeři budou moci přidávat uživatele jako zaměstnance do svých podniků.
    - FP 5.3.1: Manažeři budou moci editovat a mazat zaměstnance podniku.
  - FP 5.3: Oprávněný uživatel bude moci skenovat QR kódy
- FP 6: Správa věrnostních karet
  - FP 6.1: Manažeři budou moci vytvářet a spravovat věrnostní karty.
- FP 7: Funkcionalita QR kódů
  - FP 7.1: Aplikace bude generovat unikátní QR kódy pro každou věrnostní kartu.
  - FP 7.2: Zaměstnanci budou moci skenovat QR kódy a přidávat razítka na věrnostní karty zákazníků.

Pro lepší zobrazení funkcionalit aplikace byl vytvořen diagram případu užití, který tyto funkcionality specifikuje.



Obrázek 6 – Diagram případu užití pro mobilní aplikaci (zdroj: vlastní)

## 4.2 Nefunkční požadavky

- NP 1 – Kompatibilita
  - NP 1.1 – Aplikace bude kompatibilní s nejnovějšími verzemi operačního systému Android.
- NP 2 – Responzivita
  - NP 1.2 – Aplikace bude responzivní a optimalizovaná pro různé velikosti mobilních obrazovek.
- NP 3 – Rozšiřitelnost
  - Aplikace bude navrhována tak, aby umožnila rozšiřování a přidávání nových funkcí v budoucnu.
- NP 4 – Výkon
  - Aplikace bude navrhována tak, aby umožnila minimální spotřebu systémových zdrojů.

## 5 Vývoj mobilní aplikace

Aplikace je vytvořena s využitím .NET MAUI a díky implementaci architektury MVVM dochází k efektivnímu rozdělení tříd podle jejich funkcionality, což zajišťuje jasné oddělení logiky aplikace od uživatelského rozhraní.

Veškeré uživatelské rozhraní používané v aplikaci je rozděleno do příslušných souborů XAML, které definují strukturu a vzhled rozhraní, a do přidružených souborů C#, které obsahují logiku řídicí chování rozhraní.

### 5.1 Vzhled aplikace

Aplikace podporuje jak tmavý, tak světlý režim, přičemž ve výchozím nastavení je nastavena na systémový režim. Uživatel má možnost si režim změnit ručně v aplikaci v sekci nastavení na svém profilu.

V dnešní době je velmi rozsáhle využíván tmavý režim, proto je také tento režim hlavním, na který byla aplikace testována. Veškeré barevné kombinace byly co nejvíce testovány pro zajištění co nejlepších barevných kombinací v tomto režimu. Režim světlý, ale není o nic horší, je spousta uživatelů, kteří tento režim využívají, a proto byly barevné kombinace co nejlépe otestovány i na tento režim.

Vzhled byl také navržen tak, aby stránky byly vizuálně co nejvíce podobné. Aplikace obsahuje některé prvky, které se mírně liší, ale snaží se využívat stejné styly pro zajištění konzistentního uživatelského rozhraní.

Každá stránka obsahující větší množství obsahu má vertikální posuvník. Tento prvek zajišťuje, že na zařízeních, která nedokážou obsah zobrazit na celou výšku kvůli malé velikosti, bude obsah přesto dostupný díky posuvníku, kterým si uživatel může zobrazit zbylou část obsahu. Na stránkách podniků a profilů se využívá také horizontální posuvník, který slouží k navigaci mezi integrovanými okny.

Aplikace také disponuje mnoha dialogovými okny, která jsou implementována ve třídě *DialogService*. Jedním z nejčastěji využívaným oknem je okno pro zobrazení aktivity, které uživateli ukazuje, že aplikace zpracovává nějaký požadavek.

### 5.2 Registrace a přihlášení

#### 5.2.1 Registrace

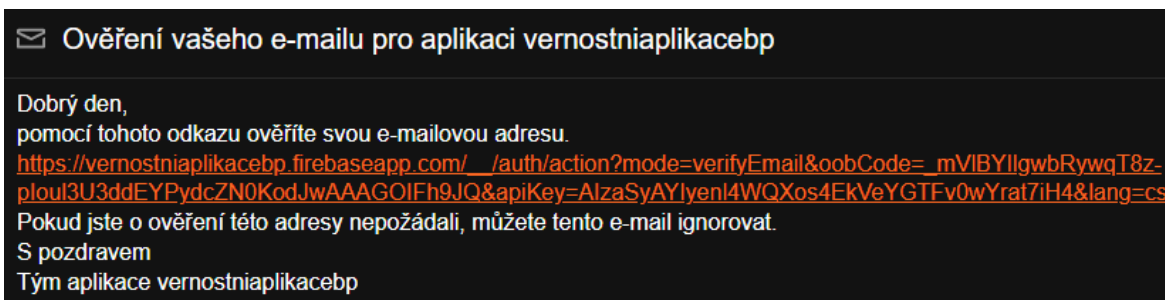
Uživatelské rozhraní registrační obrazovky se nachází ve třídě *RegistrationPage*, toto uživatelské rozhraní obsahuje vzhledovou část pro režim na šířku a režim na výšku.

The image shows a registration form titled "Registrace" on a dark background. It contains five input fields, each with a label and an asterisk indicating it is required: "Křestní jméno \*", "Příjmení \*", "Email \*", "Heslo \*", and "Potvrďte heslo \*". Below each label is a light gray input field with a placeholder text matching the label: "Křestní jméno", "Příjmení", "Email", "Heslo", and "Potvrďte heslo". At the bottom of the form is a light gray button labeled "Zaregistrovat se". Below the button is a link that says "Máte již účet? Přihlásit se".

**Obrázek 7 – Obrazovka pro registraci (zdroj: vlastní)**

Hlavní funkce registrační obrazovky jsou implementovány v třídě *RegistrationViewModel*. Před dokončením registrace je provedena validace vstupů uživatele, aby se zajistilo, že jsou všechna pole správně vyplněna. Kontrola formátu e-mailové adresy a hesla se provádí pomocí regulárních výrazů. V případě chybného vyplnění některého z polí se uživateli zobrazí chybová zpráva specifikující danou chybu.

Pro registraci uživatelů je využívána služba Firebase Authentication, která ukládá jejich základní údaje a v reálném čase zpracovává jakékoli změny. Součástí procesu je i odesílání potvrzovacího e-mailu. Uživatel musí potvrdit svoji e-mailovou adresu prostřednictvím zaslání odkazu; bez ověření mu nebude povoleno přihlášení do aplikace. Po úspěšném dokončení registrace je uživatel automaticky přesměrován na přihlašovací obrazovku.



Obrázek 8 – Příklad potvrzovací e-mailové zprávy (zdroj: vlastní)

### 5.2.2 Přihlášení

Stejně jako obrazovka pro registraci, obsahuje přihlašovací obrazovka vzhledovou část pro režim na šířku i na výšku, uživatelské rozhraní pro přihlašování se nachází ve třídě *LoginPage*.

Obrázek 9 – Obrazovka pro přihlášení (zdroj: vlastní)

Funkcionalita přihlašovací obrazovky je implementována ve třídě *LoginViewModel*. Tato třída nejprve ověří správnost uživatelských údajů. Pokud jsou údaje validní, pokračuje v ověření pomocí Firebase Authentication. Tento autentizační systém zahrnuje automaticky nastavená pravidla, jako je například omezený počet přihlašovacích pokusů v určitém časovém intervalu. V případě specifických chyb se uživateli zobrazí příslušné chybové hlášky. Součástí ověřovacího procesu je také kontrola, zda si uživatel potvrdil svou e-mailovou adresu. Jestliže e-mail nebyl potvrzen a přihlašovací údaje jsou správné, uživatel obdrží možnost na nové zaslání potvrzovacího odkazu.

Po úspěšném přihlášení se načtou dodatečné informace o uživateli z Firestore dokumentu. To je nezbytné, protože implementace Firebase Authentication, která je v aplikaci využita ukládá pouze e-mailovou adresu a heslo. Po načtení těchto údajů jsou data přidána do session, kterou spravuje *SessionManager*. Následuje načtení obecných i specifických informací o podnicích, které zpracovává *DataManager*. Tato třída udržuje a spravuje všechna relevantní data. Po úspěšném načtení a zpracování dat je uživatel přesměrován na hlavní obrazovku aplikace.

## 5.3 Správa dat a session management

### 5.3.1 Třída *SessionManager*

Tato třída zastává důležitou roli v autentizačních procesech a správě uživatelských session v aplikaci. Primárně interaguje se službou Firebase Authentication, která umožňuje správu akcí jako je přihlášení, odhlášení a ověření uživatelů. Kromě toho tato třída zajišťuje možnost změny e-mailových adres a hesel přihlášených uživatelů.

Třída také umožňuje získání tokenu Firebase aktuálně přihlášeného uživatele pro využití autentizace do cloudového úložiště. *SessionManager* rovněž spravuje a v reálném čase uchovává informace o aktuálně přihlášeném uživateli. Týká se to základních informací, jako jsou e-mail a heslo, které mohou být uživatelem aktualizovány přímo přes tuto třídu. Dodatečné informace uživatele jsou zde pouze uchovávány a nejsou přes *SessionManager* upravitelné. Pro změnu těchto informací je nutné využít třídu *UserAdditionalInfoService*, která jako několik podobných tříd slouží ke správě údajů v databázi.

### 5.3.2 Třída *DataManager*

Třída *DataManager* je jednou z nejdůležitějších a nejvíce využívaných tříd v mobilní aplikaci. Její primární úlohou je uchovávání aktuálních informací o podnicích. *DataManager* spravuje jak všeobecné podniky, tak i ty, které jsou specifické pro jednotlivé uživatele, včetně těch, kde je uživatel zaměstnancem nebo manažerem.

Klíčovou funkcionalitou této třídy je implementace listenerů, které monitorují databázi v reálném čase. Tyto listenery okamžitě detekují jakékoli změny v databázi, což umožňuje rychlé načtení a aktualizaci dat v aplikaci. *DataManager* dále spravuje i výchozí obrázky, které jsou používány jak pro profil podniků, tak pro uživatele. Díky této třídě, pokud dojde například ke změně informací o podniku, jsou tato data okamžitě aktualizována na obrazovce uživatele, který si prohlíží tento podnik, bez potřeby ručního obnovení stránky.

## 5.4 Hlavní obrazovka

Po úspěšném přihlášení je uživatel přesměrován na hlavní obrazovku, která slouží jako centrální rozhraní pro navigaci mezi různými částmi aplikace. Tato obrazovka primárně nabízí přehled podniků dostupných v aplikaci. Každý podnik je prezentován v zaobleném čtyřúhelníku, který obsahuje základní údaje jako název, adresu a krátký popis. Barevná schémata těchto obdélníků se adaptují podle režimu zobrazení, který uživatel používá.



Obrázek 10 – Příklad vzhledu karty pro jednotlivý podnik (zdroj: vlastní)

Hlavní obrazovka je vybavena systémem stránkování, což zajišťuje rychlé a efektivní zobrazování dat. Počet podniků zobrazených na jedné stránce je omezen na pět, s možností jednoduchého přechodu mezi stránkami.

Dále obrazovka poskytuje několik možností filtrace. Uživatelé mohou podniky třídit podle různých kritérií, jako je název podniku, kategorie (restaurace, kavárna, pizzerie atd.), nebo geografická poloha. Při filtraci podle polohy musí uživatel poskytnout aplikaci přístup k poloze; následně jsou zobrazeny podniky v okruhu deseti kilometrů. Filtrace může kombinovat různá kritéria, ale nelze kombinovat vyhledávání podle textu společně s vyhledáváním podle polohy.

V pravém horním rohu obrazovky se nachází profilový obrázek uživatele, který slouží jako odkaz na jeho profil.

## 5.5 Uživatelský profil

Uživatelský profil je unikátní pro každého uživatele, avšak rozvržení je stejné pro všechny. Profil obsahuje profilový obrázek, který si uživatel buď sám vybere, nebo je použit jeden z dostupných výchozích obrázků. Vedle profilového obrázku je umístěno jméno uživatele. Na této obrazovce jsou také integrovány čtyři další komponenty, mezi kterými lze snadno přepínat. Jedna komponenta zobrazuje všechny orazítkované věrnostní karty uživatele, další komponenta ukazuje podniky, kde je uživatel zaměstnancem nebo manažerem, a zbývající komponenty slouží ke změně specifických uživatelských údajů a obecného nastavení.



Obrázek 11 – Ukázka části profilu s horizontálním posuvníkem (zdroj: vlastní)



### 5.5.1 Věrnostní karty, které má uživatel orazítkované

První z integrovaných komponent na profilu uživatele zobrazuje unikátní věrnostní karty, které má uživatel alespoň z části orazítkované. Věrnostní karty jsou podobně jako podniky zobrazeny s maximálním limitem pěti věrnostních karet na stránku, s možností listování mezi stránkami. Každé razítko je zobrazeno ve svém zaobleném čtyřúhelníku, podobně jako vzhled podniků, ale s tím rozdílem, že pozadí karet je průhledné. Každá karta má definovaný počet razítek potřebných pro její úplné splnění. Potenciální razítka jsou na začátku zobrazena v prázdném, průhledném a zaobleném čtverci, který se po získání razítka vyplní příslušnou ikonou. Karta také zahrnuje informaci o platnosti, přičemž neplatné karty nejsou v této části zobrazeny. Klíčovou komponentou každé věrnostní karty je QR kód, jehož generování a skenování je podrobně popsáno v samostatné kapitole.



Obrázek 12 – Věrnostní karta obsahující razítko (zdroj: vlastní)

### 5.5.2 Podniky přihlášeného uživatele

Každý podnik vytvořený v aplikaci má jednoho manažera, který má pravomoc spravovat veškeré informace o podniku a o přidružených věrnostních kartách. Podniky mohou mít zaměstnance, které si vybírá manažer. Tito zaměstnanci nemají oprávnění upravovat informace o podniku; jedinou funkcionalitu, kterou mohou využívat, je skenování QR kódů příslušného podniku.

Na profilové obrazovce, vedle sekce věrnostních karet, je možnost přepnutí na komponentu zobrazující podniky, ve kterých je uživatel zaměstnancem nebo manažerem. Podobně jako v případě zobrazení věrnostních karet, i podniky jsou prezentovány na stránkách. Každý podnik nabízí stejně jako na hlavní obrazovce možnost prokliku na detailní obrazovku s konkrétními informacemi o vybraném podniku. Na této obrazovce je také umístěna možnost založit nový podnik, což uživateli umožní stát se jeho manažerem.

### 5.5.3 Změna údajů

Následující komponenta, dostupná na profilové obrazovce, slouží pro změnu uživatelských údajů. Uživatelé zde mají možnost vybrat si ze tří druhů změn, kterými jsou osobní údaje, avatar a heslo.

Při změně osobních údajů mohou uživatelé upravit své křestní jméno, příjmení, telefonní číslo a e-mailovou adresu. Kromě e-mailové adresy stačí pro potvrzení změn pouze stisknout tlačítko potvrdit. Změna e-mailové adresy vyžaduje dodatečné ověření heslem. Pokud e-mailová adresa již existuje, uživatel nemůže provést změnu na tuto adresu. Po úspěšné změně e-mailu je uživatel automaticky odhlášen a na novou adresu je zaslán potvrzovací e-mail, který je nezbytné potvrdit pro opětovné přihlášení.

Změna avataru nabízí uživatelům možnost vidět svůj profilový obrázek ve třech různých velikostech, aby mohli lépe posoudit, jak budou vypadat v různých částech aplikace. Uživatelé si mohou vybrat vlastní obrázek ze svého zařízení, což vyžaduje povolení přístupu k souborům, nebo mohou vybírat z výchozích obrázků. Nové výchozí obrázky může přidávat pouze administrátor.



Obrázek 13 – Zobrazení profilového obrázku s možností nahrát nový (zdroj: vlastní)

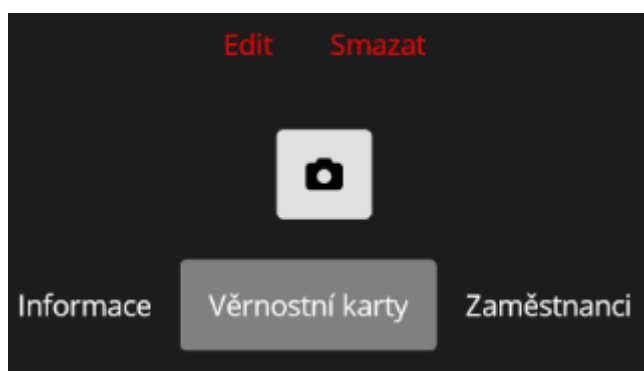
Poslední z možných úprav je změna hesla. Uživatel musí nejprve zadat stávající heslo a poté zadat nové heslo spolu s jeho potvrzením. Nové heslo podléhá stejným pravidlům jako při registraci a musí splňovat kritéria definovaná specifickým regulárním výrazem, aby byla zaručena jeho dostatečná složitost. Na rozdíl od změny e-mailové adresy, po úspěšné změně hesla nedochází k odhlášení uživatele.

### 5.5.4 Nastavení

Poslední komponenta, která je integrována do profilové obrazovky, se věnuje základním nastavením aplikace. Uživatelé zde mají možnost změnit barevný režim, přičemž mohou volit mezi světlým a tmavým režimem. Dále je v této komponentě umístěna možnost odhlášení z aplikace.

## 5.6 Obrazovka podniku

Jednou z nejdůležitějších obrazovek v aplikaci, je obrazovka určená pro konkrétní gastronomické podniky. Podobně jako profilová obrazovka, i tato obrazovka nabízí možnost přepínání mezi různými specifickými komponenty. Běžný uživatel na této obrazovce uvidí obrázek podniku spolu s možnostmi přechodu na komponenty se základními informacemi a věrnostními kartami. Manažeři podniku a uživatelé s administrátorskými právy mají navíc možnost editovat nebo smazat podnik a přepnout se na okno zobrazující seznam zaměstnanců. Každý zaměstnanec, manažer a administrátor může také skenovat QR kódy.



Obrázek 14 – Možnost skenování a úprav pro oprávněné uživatele (zdroj: vlastní)

### 5.6.1 Informace o podniku

Tato komponenta poskytuje základní informace o podniku, zahrnující název, popis, otevírací dobu, kontaktní údaje a adresu. Dále zobrazuje odkazy na sociální média, jako jsou platforma X (dříve Twitter), Facebook a Instagram.

### 5.6.2 Věrnostní karty

Věrnostní karty jsou hlavní funkcionalitou této aplikace, tato komponenta umožňuje zobrazit jak výchozí, tak unikátní věrnostní karty, které má uživatel alespoň z části orazítkované. Výchozí karty jsou neorazítkované a slouží jako nová prázdná karta. Uživatelé si mohou nechat orazítkovat jak výchozí, tak již částečně orazítkované karty. Pokud si uživatel nechá orazítkovat prázdnou novou kartu a zároveň má již částečně orazítkovanou kartu stejného typu, razítko se přidá do již orazítkované karty, dokud není kompletně vyplněná. Jakmile je karta kompletně vyplněná, začne se razítkovat nová karta.

Na této komponentě se také využívá stránkování, a běžní uživatelé vidí pouze platné karty. Manažeři a administrátoři mají přístup i k neplatným výchozím kartám, což jim umožňuje karty prodloužit nebo smazat.

Manažeři a administrátoři mají na této komponentě také možnost přidávat a editovat věrnostní karty. Obrazovka pro přidávání a editaci věrnostních karet je stejná, ale liší se podle toho, zda se přidává nová karta nebo se edituje stávající. Při přidávání karty musí manažer specifikovat popis karty, její účel, minimální platnost karty, která je nastavena na měsíc, a počet razítek potřebných k získání odměny, který může být v rozmezí od pěti do dvaceti razítek. Také si vybere ikonu pro razítko z dostupných možností. Při editaci může

změnit popis, platnost a ikonu razítka, ale počet razítek není možné upravit, v takovém případě je nutné vytvořit novou věrnostní kartu.

### 5.6.3 Zaměstnanci

Komponenta zaměstnanců je přístupná výhradně manažerům podniku a administrátorům. V této komponentě jsou zobrazeni všichni zaměstnanci včetně manažera. Každý podnik má pouze jednoho manažera, jehož role je neměnná. Komponenta obsahuje funkce pro přidávání a editaci zaměstnanců přímo, bez potřeby použití externí obrazovky. Manažer nebo administrátor může mezi režimy zobrazení zaměstnanců přepínat. Existuje klasické zobrazení pro prostý náhled a editační zobrazení, které umožňuje přidávat a upravovat informace o zaměstnancích daného podniku přímo v tomto okně.

### 5.6.4 Přidávání a editace podniku

Obrazovka sloužící k editaci podniků je zároveň využívána i pro jejich přidávání. Uživatelé mohou na svém profilu v sekci podniků přidávat nové podniky. Pokud mají dostatečná oprávnění, mohou rovněž na obrazovce konkrétního podniku provádět editaci informací. Proces editace a přidání zahrnuje řadu textových a specifických polí, jejichž vstupy jsou pečlivě validovány a kontrolovány skrze kód. Zadaná adresa je převáděna na geografické souřadnice pomocí API, které umožňuje jejich získání ze zadané adresy. V případě, že zadaná adresa není rozpoznatelná a nebylo možné najít její geografické souřadnice, uživatel dostane chybovou hlášku a musí adresu upravit. Jakmile jsou všechna pole správně vyplněna a adresní údaje ověřeny, dochází k aktualizaci nebo vytvoření záznamu o podniku v databázi.

## 5.7 Využití služby Firebase v mobilní aplikaci

V mobilní aplikaci je využíváno několik cloudových služeb od Firebase. Využívané služby jsou Firebase Authentication, Firebase Cloud Storage a Cloud Firestore.

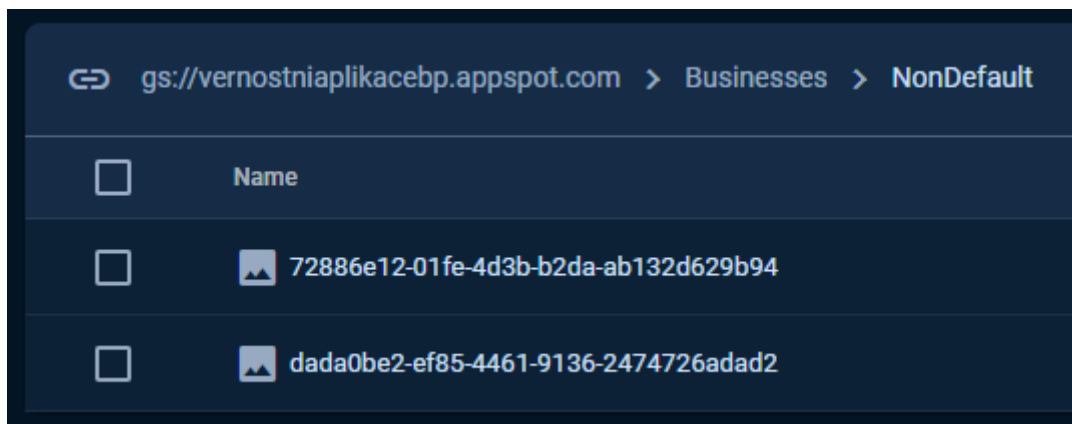
### 5.7.1 Firebase Authentication v aplikaci

Autentizace od Firebase je využívána pro jednoduchou správu přihlašování uživatelů v mobilní aplikaci. Firebase Authentication obsahuje spoustu funkcionalit, které lze jednoduše využít bez sebemenších problémů. Aplikace mimo správu přihlašování zároveň využívá tuto službu na posílání e-mailových zpráv při nové registraci nebo změně e-mailové adresy. Příklad ukázky zaslaného potvrzovacího e-mailu je na přiloženém obrázku „Obrázek 8,,.

### 5.7.2 Firebase Cloud Storage v aplikaci

Cloud Storage od Firebase je využit jako cloudové úložiště využívaných obrázků v aplikaci. Úložiště uchovává obrázky všech podniků a uživatelů. Veškeré výchozí a unikátní obrázky, které jsou určeny jak pro podnik, tak uživatele jsou zde uchovávány. Třída *FirebaseStorageService* obsahuje veškerou funkcionalitu pro nahrávání souborů a jejich správu. Všechny metody v této třídě využívají asynchronní přístup a pro možnost získání dat nebo jakoukoliv správu souborů v úložišti je potřeba využít Firebase token, který je získán od aktuálně přihlášeného uživatele a díky němu je možnost využívat těchto

funkcionalit. Každý nový obrázek má vygenerovaný unikátní název, tento název se později využívá pro nahrazování, díky unikátnímu názvu není potřeba mazat starý obrázek, protože nový obrázek nahradí obrázek původní.



Obrázek 15 – Firebase Cloud Storage s ukázkou unikátních názvů (zdroj: vlastní)

### 5.7.3 Cloud Firestore v aplikaci

Služba Cloud Firestore je v aplikaci používána k správě dat týkajících se podniků, uživatelů, věrnostních karet a obrázků. Třídy určené pro serializaci dat z a do Firestore jsou označeny atributem `[FirestoreData]`, který říká, že daná třída odpovídá struktuře dokumentu v NoSQL databázi. Aby bylo možné v dokumentu použít specifické vlastnosti, je nutné je označit atributem `[FirestoreProperty]`. Tento atribut umožňuje mapování vlastností na pole v dokumentu. Vlastnosti bez tohoto atributu jsou při serializaci přeskočeny.

Před použitím Cloud Firestore je nezbytné aplikaci povolit přístup k službám od společnosti Google. Přístup je řízen pomocí specifických autentizačních údajů, které by měly být v produkčním prostředí bezpečně uloženy a chráněny před neoprávněným přístupem. Tyto autentizační údaje jsou klíčem k používání cloudových služeb Firebase.

V aplikaci jsou dále implementovány třídy jako *BusinessService*, *ImageService*, *UserAdditionalInfoService* a *UserLoyaltyCardService*, které spravují specifické kolekce v databázi. Každá z těchto tříd nabízí metody a funkce specifické pro správu příslušných kolekcí v databázi.

## 5.8 Generování a skenování QR kódů

Jednou z hlavních funkcionalit této aplikace je práce s QR kódy, zejména jejich generování a skenování. K tomuto účelu je použita knihovna ZXing a její verze, která je speciálně navržena pro .NET MAUI. Každý QR kód generovaný v aplikaci je unikátní a obsahuje specifické informace pro danou věrnostní kartu a uživatele. Tyto kódy uchovávají informace jako identifikátory podniku, věrnostní karty a uživatele. Zároveň karta obsahuje časové razítko, které později slouží ke kontrole platnosti QR kódu. V případě, že je již karta orazítkována tak obsahuje také identifikátor orazítkované karty.

Z bezpečnostních důvodů jsou veškerá data v QR kódu šifrována pomocí symetrické šifry AES. Symetrické šifry používají stejný klíč jak pro šifrování, tak pro dešifrování dat. V této aplikaci se pro každý podnik používá unikátní šifrovací klíč, který je bezpečně uchováván v databázi. Jak šifrovací klíč, tak šifrovaná data jsou kódována pomocí Base64.

```
private string EncryptData(string data, string base64Key)
{
    using (Aes aesAlg = Aes.Create())
    {
        aesAlg.Key = Convert.FromBase64String(base64Key);

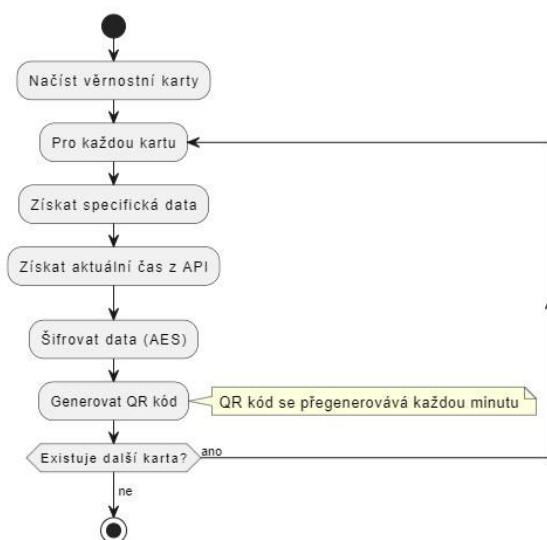
        aesAlg.IV = new byte[16];

        ICryptoTransform encryptor = aesAlg.CreateEncryptor(aesAlg.Key, aesAlg.IV);

        using (MemoryStream msEncrypt = new MemoryStream())
        {
            using (CryptoStream csEncrypt = new CryptoStream(msEncrypt, encryptor, CryptoStreamMode.Write))
            {
                using (StreamWriter swEncrypt = new StreamWriter(csEncrypt))
                {
                    swEncrypt.Write(data);
                }
            }
            return Convert.ToBase64String(msEncrypt.ToArray());
        }
    }
}
```

Obrázek 16 – Metoda využívaná pro zašifrování dat pomocí šifry AES (zdroj: vlastní)

QR kódy se generují na dvou místech v aplikaci, a to na profilové obrazovce v sekci pro uživatelské věrnostní karty, kde jsou zobrazeny orazítkované věrnostní karty uživatele, a na kartě každého podniku v sekci věrnostních karet. Generování QR kódů se automaticky obnovuje každou minutu. Aby bylo zajištěno, že čas použitý v aplikaci je vždy přesný, využívá se API pro zjištění aktuálního času. Toto opatření zabraňuje využití nesprávného systémového času, který by mohl být na uživatelově zařízení nesprávně nastaven. Tato funkcionality je implementována ve třídě *TimeService*.



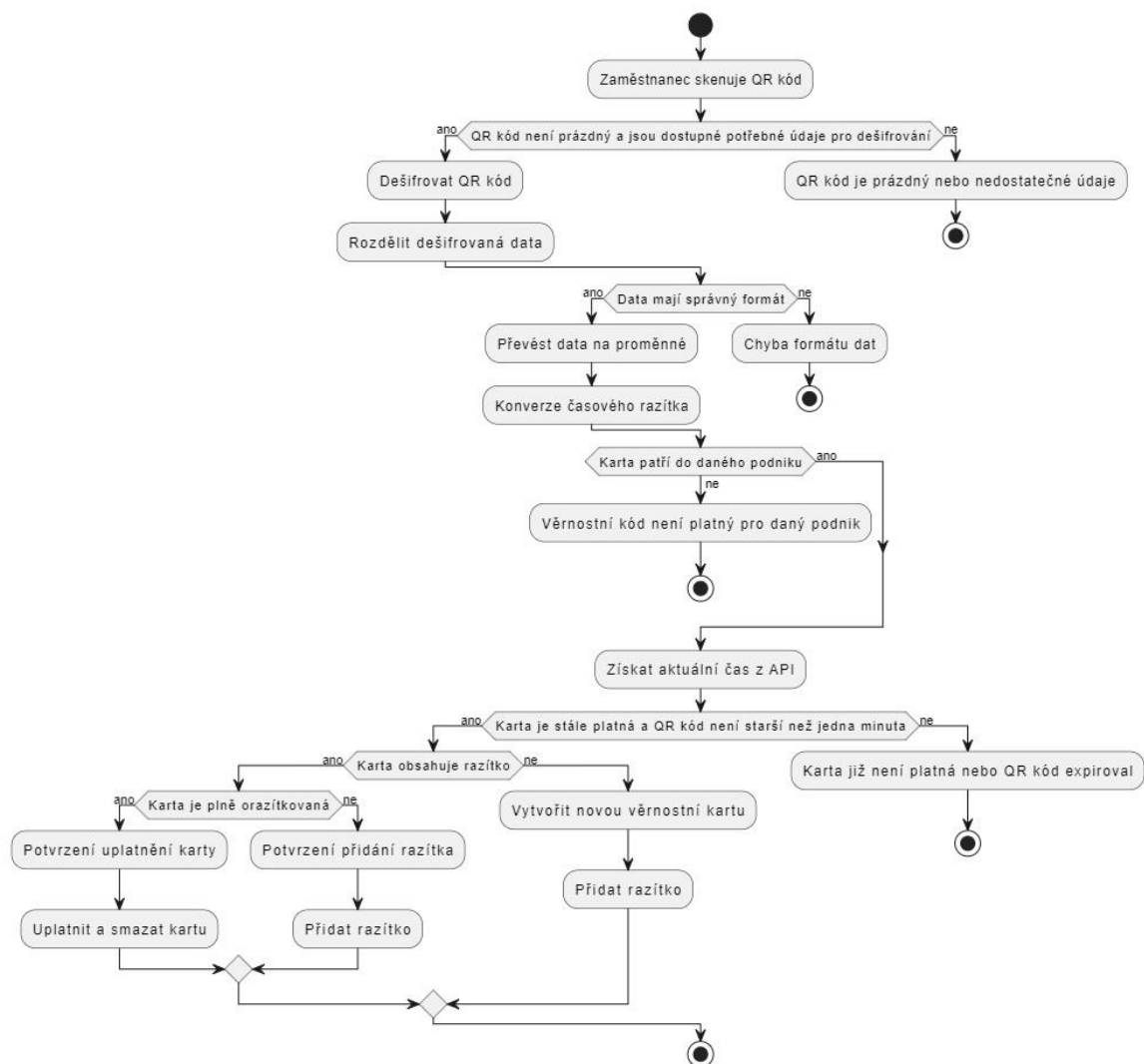
Obrázek 17 – Diagram aktivit ukazující průběh generování QR kódů (zdroj: vlastní)

Generované QR kódy mohou na první pohled vypadat malé a těžko skenovatelné, ale každý z nich je možné rozkliknout, což uživateli zobrazí zvětšenou verzi kódu v popředí. I v této zvětšené formě se QR kódy pravidelně obnovují, což zajišťuje, že ke skenování je vždy použit pouze nejaktuálnější kód.



**Obrázek 18 – Zvětšený, vygenerovaný QR kód (zdroj: vlastní)**

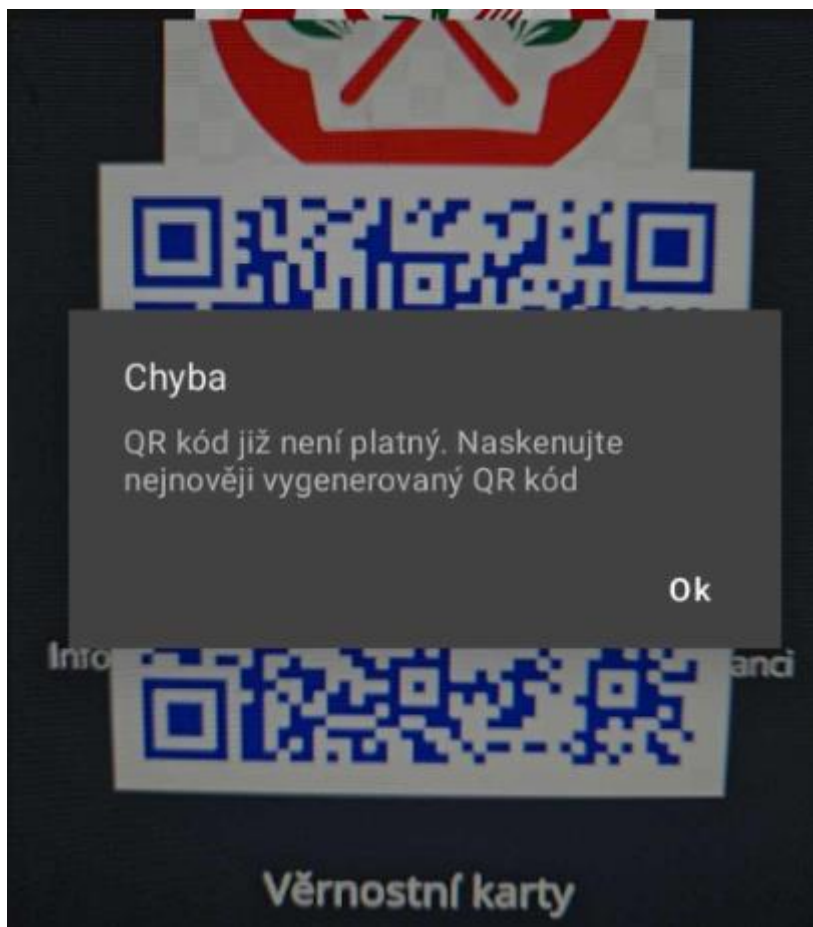
Skenování QR kódů se provádí na obrazovce konkrétních podniků, a je dostupné pouze uživatelům s příslušnými oprávněními. Systém obsahuje důkladné kontroly, které zajišťují, že lze naskenovat pouze QR kódy odpovídající správnému formátu a příslušnému podniku. Každý podnik je vybaven speciálním skenerem, který umožňuje skenování pouze těch QR kódů, které jsou mu přiřazeny.



**Obrázek 19 – Diagram aktivit ukazující průběh skenování QR kódů (zdroj: vlastní)**

Pokud se oprávněný uživatel pokusí naskenovat QR kód nesouvisející s daným podnikem, systém vyvolá chybovou hlášku a skenování se přeruší. Může se též stát, že skenovaný QR kód patřící danému podniku již vypršel, což je dáno tím, že QR kódy mají omezenou platnost jedné minuty. V takovém případě se zobrazí chybová hláška a je potřeba naskenovat nově vygenerovaný QR kód.





**Obrázek 20 – Chybová hláška pro zneplatněný QR kód (zdroj: vlastní)**

Skenování QR kódů je možné pouze tehdy, když uživatel aplikaci udělí oprávnění ke kameře. Po získání tohoto oprávnění může oprávněný uživatel využívat funkci skenování. V případě, že je naskenován platný a správný QR kód, je uživateli, jehož kód to je, přiděleno razítko.

## Závěr

Cílem této práce bylo vytvoření věrnostní mobilní aplikace, jejíž hlavní funkcionalita je založena na QR kódech. V úvodní části byly popsány věrnostní karty a jejich existující alternativy, stejně jako věrnostní systémy úzce spojené s věrnostními kartami. Další kapitoly teoretické části se věnovaly existujícím alternativám klasických věrnostních karet na českém trhu, technologii QR kódů a technologiím použitým ve vývoji.

Praktická část se zaměřila na návrh a analýzu funkčních a nefunkčních požadavků pro mobilní aplikaci a na znázornění funkčních požadavků v diagramu případů užití. Tato část práce se dále věnovala vývoji zmíněné mobilní aplikace určené pro operační systém Android. Byly postupně popsány všechny funkcionality, od klasické registrace a přihlášení, přes správu profilů a podniků, využívání různých API pro získávání času a geografických souřadnic, až po implementaci hlavní funkcionality QR kódů pro správu věrnostních karet. Funkcionalita generování a skenování QR kódů byla podrobněji popsána pomocí diagramů aktivit, které znázorňují průběh těchto funkcionalit.

Aplikace implementuje veškeré specifikované funkční požadavky a snaží se co nejvíce dodržovat požadavky nefunkční. Aplikace byla také testována na několika zařízeních s operačním systémem Android a byl připraven základ pro operační systém iOS, který nebyl otestován z hlediska nedostatku testovacích zařízení pro tento operační systém.

Aplikace intenzivně využívá cloudové služby jak pro správu obrázků a autentizaci, tak pro správu dat prostřednictvím NoSQL databáze. Google Firebase se ukázal jako skvělé řešení pro integraci cloudových služeb do mobilních aplikací, jelikož nabízí široké portfolio služeb a nepožaduje zadávání platebních údajů, což bylo jedním z hlavních důvodů pro výběr této platformy.

Během vývoje bylo zjištěno, že framework .NET MAUI obsahuje mnoho chyb, které jsou sice komunitou postupně opravovány, ale do doby opravy je nutné najít alternativní řešení těchto problémů, což vyžaduje větší využití času. Přesto .NET MAUI nabízí výhodu v podobě možnosti psát jednotný kód pro mnoho operačních systémů, což zjednodušuje testování aplikace na platformě Windows, i když je aplikace určena pro mobilní zařízení. Tato platformní nezávislost umožnila provádět základní testování funkcionalit přímo na Windows, aniž by bylo nutné používat virtuální telefony nebo fyzická zařízení. Tyto zařízení byly využity až pro testování pokročilejších funkcí aplikace.

Pro budoucí rozvoj aplikace se nabízí možnost rozšíření podpory i pro operační systém iOS a implementace alternativních možností přihlášení a registrace prostřednictvím jiných poskytovatelů, jako je Google. Toto by mohlo přilákat širší uživatelskou základnu a zvýšit flexibilitu používání aplikace. Kromě toho je klíčové vyvinout funkcionality, které umožní monitorování a schvalování zakládání nových podniků v aplikaci. Tím by se zajistilo, že podnik může být založen pouze uživatelem, který obdržel předchozí schválení od správce aplikace.

## Literatura

- [1] TURNER, Ash. *Android vs. Apple Market Share: Leading Mobile Operating Systems (OS)* (2024). [online] 2024 [cit. 2024-04-16]. Dostupné z: <https://www.bankmycell.com/blog/android-vs-apple-market-share/>
- [2] StatCounter. *Mobile Operating System Market Share Czech Republic*. [online] 2024 [cit. 2024-04-16]. Dostupné z: <https://gs.statcounter.com/os-market-share/mobile/czech-republic>
- [3] Shopuddy. *Digital loyalty card VS Physical loyalty Card (Stamp Card, Membership Card)* 2023. [online] 2022 [cit. 2024-05-01]. Dostupné z: <https://www.shopuddy.com/en/blog/digital-loyalty-card-vs-physical-loyalty-card>
- [4] TOMES, Paul. *Loyalty Cards for Business in 2024: Plastic vs Digital Cards – Which is Better?*. [online] 2022 [cit. 2024-05-01]. Dostupné z: <https://passkit.com/blog/loyalty-cards-for-business/>
- [5] JONES, Alexandra. *What is a loyalty program, and how can it work for your restaurant?* [online] 2024 [cit. 2024-04-19]. Dostupné z: <https://restaurant.opentable.com/resources/what-is-a-loyalty-program/>
- [6] BLACK, Halona. *Loyalty programs: How they work, tips, and examples*. [online] 2024 [cit. 2024-04-30]. Dostupné z: <https://www.zendesk.com/blog/loyalty-rewards/>
- [7] Restolabs Team. *Pros And Cons Of Reward Programs For Restaurants*. [online] 2023 [cit. 2024-04-19]. Dostupné z: <https://www.restolabs.com/blog/pros-and-cons-restaurant-loyalty-programs>
- [8] Stammgast. *Stammgast*. [online] c2021 [cit. 2024-04-19]. Dostupné z: <https://www.stammgast.cz/>
- [9] SmartBrains solutions s.r.o. *Stammgast: odměny za věrnost*. [online] 2023 [cit. 2024-04-30]. Dostupné z: <https://apps.apple.com/cz/app/stammgast-odm%C4%9Bny-za-v%C4%9Brnost/id1502892146>
- [10] Con Gusto. *Con Gusto*. [online] c2024 [cit. 2024-04-30]. Dostupné z: <https://www.congusto.cz/>
- [11] Con Gusto. *Vývoj věrnostního systému pro restaurace Con Gusto*. [online] 2022 [cit. 2024-04-19]. Dostupné z: <https://www.cognito.cz/reference/vyvoj-vernostniho-systemu-pro-restaurace-con-gusto>

- [12] Qerko. *Qerko*. [online] (bez data) [cit. 2024-04-19]. Dostupné z: <https://www.qerko.com/cs>
- [13] LÁSKOVÁ, Tereza. *Věrnostní program Qerko od začátku roku využilo 70 % hostů restaurací a kaváren*. [online] 2022 [cit. 2024-04-30]. Dostupné z: <https://fzone.cz/clanky/vernostni-program-qerko-od-zacatku-roku-vyuzilo-70-hostu-restauraci-a-kavaren-4210>
- [14] BOHUMSKÁ, Markéta. *Qerko je jen ta nejpohodlnější platba na světě. A mnohem víc*. [online] 2023 [cit. 2024-04-30]. Dostupné z: <https://www.qerko.com/cs-blog/qerko-neni-uz-davno-jen-ta-nejpohodlnejsi-platba-na-svete>
- [15] HAYES, Adam. *Quick Response (QR) Code: Definition and How QR Codes Work*. [online] 2024 [cit. 2024-04-18]. Dostupné z: <https://www.investopedia.com/terms/q/quick-response-qr-code.asp>
- [16] Kaspersky Lab. *QR Code Security: What are QR codes and are they safe to use?* [online] c2024 [cit. 2024-04-30]. Dostupné z: <https://www.kaspersky.com/resource-center/definitions/what-is-a-qr-code-how-to-scan>
- [17] Microsoft. *Prohlídka jazyka C#*. [online] 2024 [cit. 2024-04-17]. Dostupné z: <https://learn.microsoft.com/cs-cz/dotnet/csharp/tour-of-csharp/>
- [18] Microsoft. *XAML*. [online] 2024 [cit. 2024-04-28]. Dostupné z: <https://learn.microsoft.com/cs-cz/dotnet/maui/xaml/?view=net-maui-8.0>
- [19] Microsoft. *Model-View-ViewModel (MVVM)*. [online] 2023 [cit. 2024-04-28]. Dostupné z: <https://learn.microsoft.com/cs-cz/dotnet/architecture/maui/mvvm>
- [20] Microsoft. *Co je .NET MAUI?*. [online] 2024 [cit. 2024-04-17]. Dostupné z: <https://learn.microsoft.com/cs-cz/dotnet/maui/what-is-maui?view=net-maui-8.0>
- [21] Microsoft. *Co je Visual Studio?*. [online] 2023 [cit. 2024-04-17]. Dostupné z: <https://learn.microsoft.com/cs-cz/visualstudio/get-started/visual-studio-ide?view=vs-2022>
- [22] HANNA, Katie Terell a Linda ROSENCRANCE . *Google Firebase*. [online] 2023 [cit. 2024-04-17]. Dostupné z: <https://www.techtarget.com/searchmobilecomputing/definition/Google-Firebase>
- [23] Google for Developers. *Firebase pricing plans*. [online] 2024 [cit. 2024-05-01]. Dostupné z: <https://firebase.google.com/docs/projects/billing/firebase-pricing-plans>

- [24] Google for Developers. *Firebase Authentication*. [online] 2024 [cit. 2024-04-17]. Dostupné z: <https://firebase.google.com/docs/auth>
- [25] Google. *Customize account management emails and SMS messages*. [online] c2024 [cit. 2024-05-01]. Dostupné z: <https://support.google.com/firebase/answer/7000714>
- [26] Google for Developers. *Cloud Storage for Firebase*. [online] 2024 [cit. 2024-04-17]. Dostupné z: <https://firebase.google.com/docs/storage>
- [27] Google for Developers. *Cloud Firestore*. [online] 2024 [cit. 2024-04-17]. Dostupné z: <https://firebase.google.com/docs/firestore>